Exercise-set 2, Algorithms and Networks 2012/2013

Please hand in the solutions on paper to Hans Bodlaender on or before the deadline, i.e., Monday, October 1, 2012.

You can write either in English or in Dutch. Your work should be well phrased, readable, understandable, and of course, correct.

Work that is hard to read or very messy will be graded with a 0. Do not forget your name and student number on your work.

1. Easy start question (1 point) Suppose we have a directed graph $G = (V, A)$ with $3n \log n$ arcs. Arcs can have arbitrary lengths, i.e., the lengths can also be negative. In how much time (in O-notation) can we solve the all pairs shortest paths problem on such graphs? (Your answer can be short; it is sufficient here to give the time bound and refer to the result proven in the course that shows it.)

2. Karp's minimum mean-weight cycle algorithm. 5 points After exercise 24-5 from *Introduction to Algorithms*, 2nd edition. Cormel, Leiserson, Rivest, Stein.

Suppose we have a directed graph $G = (V, A)$, and each arc $e \in A$ has a weight $w(e) \in \mathbf{R}$. Write $n = |V|$ and $m = |A|$.

The **mean weight** of a cycle c with successive arcs $e_1, e_2, \ldots, e_r$ equals

$$\mu(c) = \frac{1}{r} \sum_{i=1}^{r} w(e_i)$$

We want to find a cycle with minimum mean weight. I.e., the minimum over all cycles c in G is denoted μ^*, and a cycle c with $\mu(c) = \mu^*$ is called a *minimum mean weight cycle* in G. In this exercise, we will design an algorithm to compute μ^*. (A small variation of the algorithm computes a minimum mean weight cycle.)

In this exercise, we assume that G is strongly connected, i.e., there is a path from each vertex v to each vertex w in G. (If G is not strongly connected, then we can split G in linear time in strongly connected components, and run the algorithm separately on each strongly connected component.)

For vertices $v, w \in V$, let $d(v, w)$ denote the minimum weight of a path from v to w in G (the *distance* in G, as usual.)

Let $d_k(v, w)$ denote the minimum weight of a shortest path from v to w that uses *exactly* k arcs. If there is no such path, then we set $d_k(v, w) = \infty$.

Let $s \in V$ be some arbitrary vertex in V.

Many of the steps below are almost trivial; some are not. Many steps use results from earlier steps. If you cannot prove a step, you can use the result still in later steps.

1. Show that if $\mu^* = 0$, then G has no negative weight cycle.

2. Show that if $\mu^* = 0$, then for all $v \in V$:
$$d(s,v) = \min_{0 \le k \le n-1} d_k(s,v)$$

3. Show that if $\mu^* = 0$, then for all $v \in V$:
$$\max_{0 \le k \le n-1} (d_n(s,v) - d_k(s,v)) \ge 0$$

4. Show that if $\mu^* = 0$, then for all $v \in V$:
$$\max_{0 \le k \le n-1} \frac{d_n(s,v) - d_k(s,v)}{n-k} \ge 0$$

5. Let c be a cycle with total weight 0. Let u and v be two vertices on c. Suppose $\mu^* = 0$. Suppose that the weight of the path from u to v along the cycle c is x. Show that $d(s,v) \le d(s,u) + x$. (Hint for this and the next part: the weight of the path from v to u along cycle c is $-x$.)

6. Let c be a cycle with total weight 0. Let u and v be two vertices on c. Suppose $\mu^* = 0$. Suppose that the weight of the path from u to v along the cycle c is x. Show that $d(s,v) \ge d(s,u) + x$. Conclude that $d(s,v) = d(s,u) + x$.

7. Suppose $\mu^* = 0$. Show that on each minimum mean weight cycle, there exists a vertex v, with
$$d(s,v) = d_n(s,v)$$
Hint: we want to find a vertex on c with a shortest path with exactly n arcs. Show that if we have a shortest path to some vertex w on c, then we can find a shortest path to some vertex w' on c that uses one arc more with w' the vertex after w on the cycle.

8. Suppose $\mu^* = 0$. Show that on each minimum mean weight cycle, there exists a vertex v, with
$$\max_{0 \le k \le n-1} \frac{d_n(s,v) - d_k(s,v)}{n-k} = 0$$

9. Suppose $\mu^* = 0$. Show that

$$\min_{v \in V} \max_{0 \le k \le n-1} \frac{d_n(s,v) - d_k(s,v)}{n-k} = 0$$

10. Show that if we add a constant t to the weight of each arc in G, then μ^* is increased by t.

11. Show that

$$\mu^* = \min_{v \in V} \max_{0 \le k \le n-1} \frac{d_n(s,v) - d_k(s,v)}{n-k}$$

12. Show that we can compute in $O(nm)$ time all values $d_k(s,v)$, for all $v \in V$, and all $k \in \{0, 1, 2, \ldots, n\}$. (Hint: use dynamic programming.)

13. Give an algorithm that computes μ^* in $O(nm)$ time.

3. Modelling as path problem (3 points) (*Exercise version by Peter van den Berg.*) If a music band wants to record an album, this has to be done in a recording studio.There are several different recording studios, say n, and we assume that they differ in two dimensions:

- *Quality.* Studios have a different quality in recording, mastering, etc. The quality of a studio is expressed as an integer number q. We assume that all recording studios have a different quality, and thus, we order the studios with respect to increasing qualities: $q_1 < q_2 < q_3 < \cdots < q_{n-1} < q_n$. I.e., studio i has quality q_i.

- *Price.* To record music at a studio, one has to pay a starting price f, and a rate per hour c. If studio i is used for l_i hours, then one has to pay $f_i + c_i \cdots l_i$; if studio i is not used, then we have to pay 0 (of course) to this studio. It is known that $f_i \ge 0$ and $c_i \ge 0$.

A record company has contracts with a large number of bands. These bands are of different quality and popularity; some are local bands and some are world stars. If the studio wants to record an album for a band, then a studio of a minimum quality is needed for that band. More precisely, the company has m bands under contract; these are numbered $1, 2, \ldots, m$. Band j requires that their album is recorded at a studio of quality at least q'_j. It is possible to record an album at a studio of better quality, however. The time needed for the recording of the album(s) of band j is called l_j. We sort the bands with respect to the requested quality, and also assume that all these are different, i.e., $q'_1 < q'_2 < \cdots < q'_m$.

The record company wants to decide which studios to use (and which not) to record all albums such that the requirements of quality are fulfilled, with minimum total price.

Show how this problem can be modelled as a shortest paths problem.

(Give a clear explanation, and also argue correctness of the model. An example or a diagram is appreciated, but you should also explain the mentiod in text. Hint: look at the inspection example handled in the course.)

4. Extra question. One point Modify the algorithm for mean-weight cycles such that it also **outputs** a cycle of minimum mean weight. It should use still $O(nm)$ time.